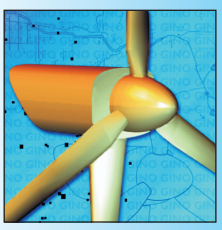
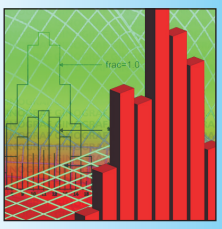
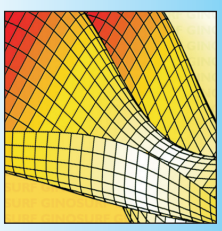
SPRING 2002

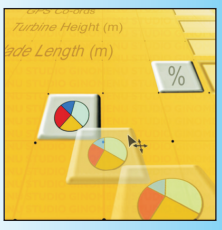
BRADLY Associates Ltd

GINO News.....



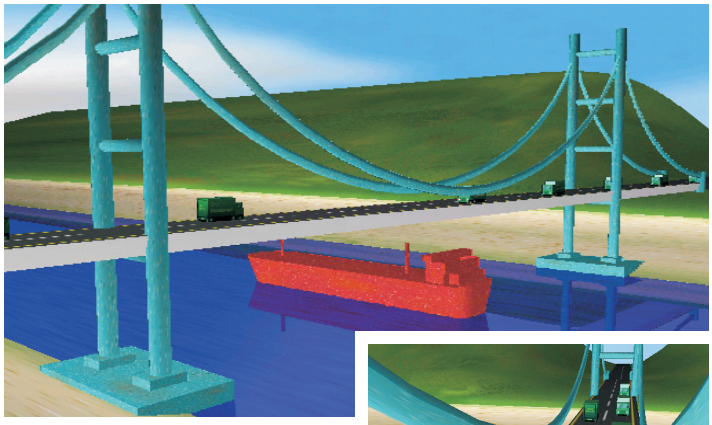




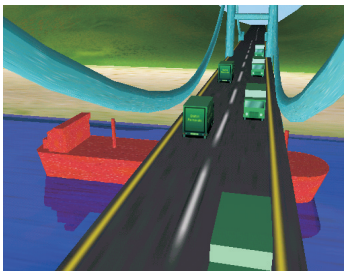


GINO Bundle v5.5 Release

Building on the success of v5.0 of the GINO suite of products, Bradly Associates is pleased to announce the release of v5.5. Many of the enhancements have been user led, resulting in a more usable product.



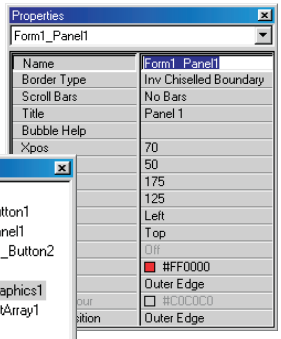
With new routines covering file handling system functions all the way through to OpenGL shading and shadows, the core GINO package has responded to user comments after the major release of v5.0. GINOMENU has added to its widget library and increased control over existing widgets while GINOSURF increases its use of triangulation to provide many more display options.



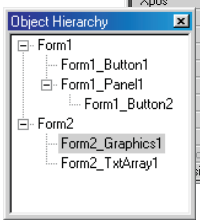
More information on new functionality is described package by package within this edition of GINO News.

GINOMENU STUDIO v3.0

GINOMENU Studio has undergone a massive overhaul to its user interface to improve its look and feel and increase the speed of application design.



As well as its existing support for Fortran 77 and 90, applications can now be generated in C/C++. Its widget set has also been increased to take advantage of the changes brought into GINOMENU v5.0.



More information is included within.

CASE STUDY Band, Lavis and Associates

(Picture courtesy of QinetiQ)

Band, Lavis and Associates, based in Maryland, USA, specializes in marine engineering and naval architecture services. Inside, Jeff Cullina, an Ocean Engineer with the company, writes about the use of the C/C++ versions of all four GINO libraries in the design of ships using parametric analysis.

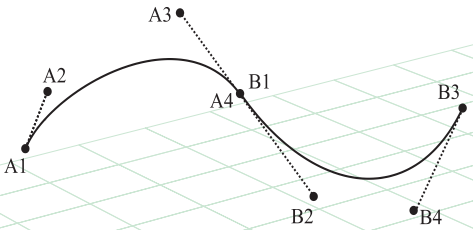
GINO graphics

GINO-F / GINO-C *v5.5*

2D and 3D Beziér Curves

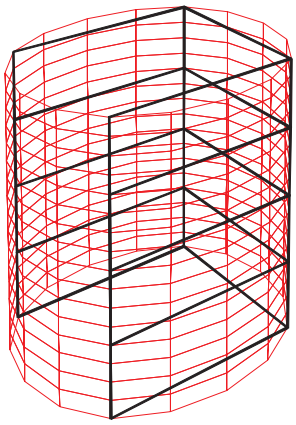
In addition to the curves currently supplied in GINO, the Beziér curve has been introduced. This curve is mostly seen in popular interactive drawing packages. It differs from the other curves in that it does not go through all the supplied control points.

The Beziér curve only goes through the start and end points with the second point forming the tangent of the curve at the start point and the penultimate point determining the tangent at the end point.



Interactive drawing packages use a series of Beziér curves with four control points with the last point of the first curve being the first point of the last. GINO will allow curves with more than four points although they are more difficult to control. Also included are commands to allow the number of control points to be increased without changing the curve. The reverse can also be done but the curve may change.

3D Spline Surface



Beziér surfaces were introduced into GINO v5.0 but as users were more used to dealing with the pre-existing spline curve, this has been expanded to surfaces that can be shaded and displayed using OpenGL.

OpenGL Planar Shadows

A limited shadow simulation has been added to the OpenGL functionality. It allows a flat shadow to be overlayed over a surface used as a floor in your scene to add extra realism.

Shaded Polylines & Facets

Although polylines could be added to your scene in v5.0, there was no control over their lighting and shading. They will now be shaded the same way as facets, if required by the user. The facets themselves could be Gouraud shaded or textured, but it was soon realised that facets should be able to be textured and Gouraud shading at the same time, so this has been added to v5.5.

Postscript Window Dumps

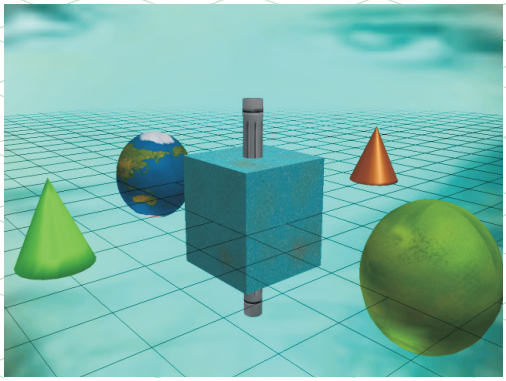
When you have an OpenGL application running, all graphic output is controlled via your video card. This means that printing using the normal methods is not possible. Two commands have been introduced to allow screen dumps to be done in Postscript from a GLX window (OpenGL X-Window) or a regular X-Window.

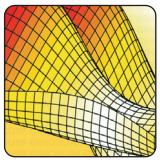
Mouse Wheel

Until now, implementation of the mouse wheel was not possible due to the lack of a standard interface. This has now been done and the mouse wheel is now an event type.

File and Directory Routines

Due to user requests, several routines have been added to allow the handling and creation of files and directories. This is added to simplify the creation of portable applications.





GINOSURF *v5.5*

Whilst there are only six new routines that have been added to the new release of GINOSURF, there are some major new display formats that have been added to the library at this time.

The most significant of these is the projected surface, generated directly from triangulated data. There are two forms to this new display type, the colour filled and the contoured filled, represented by the following two new routines:

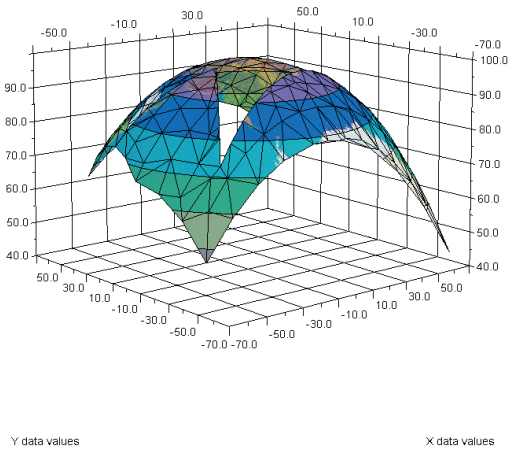
gsFillTriangulatedSurface()
gsFillTriangulatedContourSurf3D()

Apart from the advantage of being able to generate surfaces directly from users data (without necessarily having to interpolate onto a grid), the new format permits both a non rectangular boundary and cut away sections for the first time in surface plots.

A useful new routine that can be used with the above is:

gsAddSurfaceNetwork()

which will overlay the coloured or contour surface with the triangulation network grid if this is desired.



As with the gridded surface display, triangulated surfaces may be shaded according to current GINO lighting conditions and may have a texture map overlayed or blended with the contour information for a comprehensive display of multi-dimensional data.

The second major new display type added to the new library, is the perspective cross-section, where a simple gridded or a complex polyline cross-section may be drawn with respect to the current 3D perspective axes. This allows the display of cross-sections in 3D for the first time in GINOSURF. This new feature does not require any new routines but is provided through the existing cross section routines:

gsDrawGriddedCrossSection()
gsDrawPolylineCrossSection()
gsSetSurf3DState()

A further enhancement to the library is provided through the new routine:

gsSetContourKeyBoxPos()

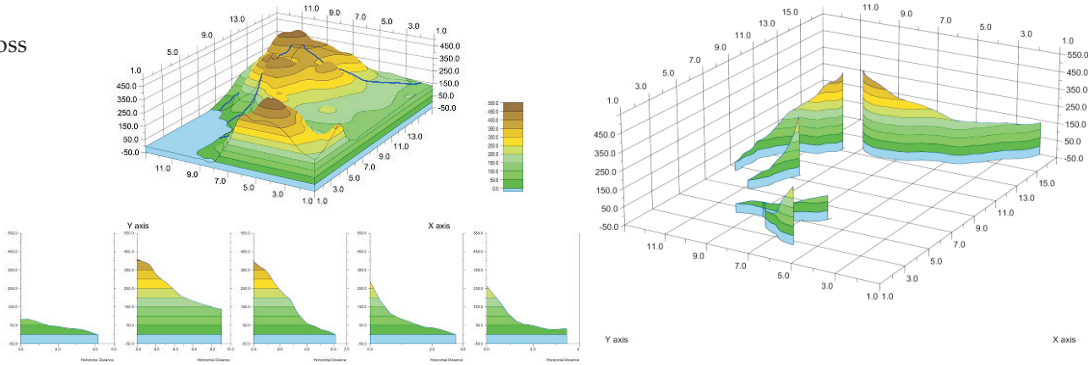
which allows a GINOSURF application to control the position, and other settings of the contour heights key box.

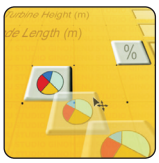
It has already been mentioned that GINOSURF can be used in conjunction with GINO's lighting and shading facilities to provide realistic impressions of surface terrains according to a wide range of lighting conditions. The final addition to the new library is a routine to control whether additional surface features such as roads, paths rivers etc. drawn using gsAddSurfacePolylineHidden() are subject to the lighting conditions or not.

The new routine is:

gsSetSurfaceLineShading()

which will in a future release also control contour lines and 3D axes drawn on shaded surfaces.





GINOMENU Studio *v3.0*

New look and feel

The look and feel of GINOMENU Studio has been completely overhauled. The property dialogue box been replaced by a scrollable property list allowing easy changes to be made to widgets. The widgets themselves can be referenced via a tree view that shows the hierarchy allowing improved project management.

Editing a widgets size and position is now easier with each widget retaining its look while it has resize corners, rather than changing to a generic editing mode view. Widget can also be selected as a group and edited "en masse". The movement and sizing of these widgets can also be snapped to a user-definable grid.

Code Generation & Compilation

Studio can now automatically generate C/C++ code as well as F77 & F90. An entire application can be created with large sections of imported code, using the enhanced project options, and then compiled and executed using the currently available Fortran or C compiler.

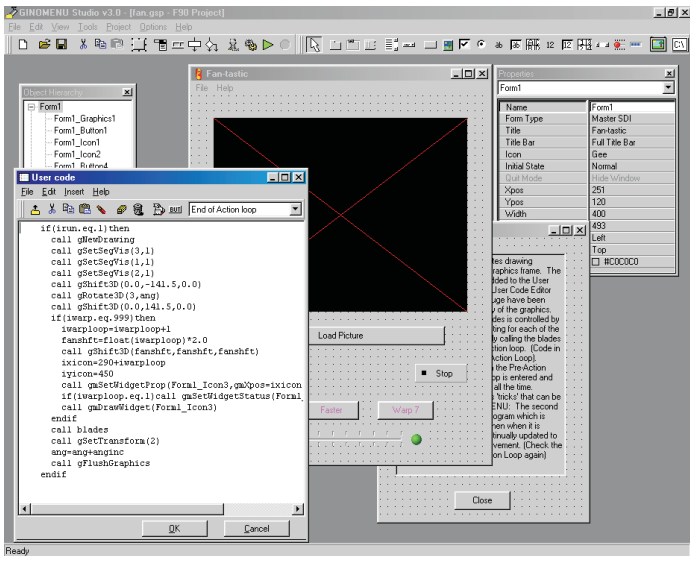
Debugging

When an application is compiled, any compilation errors are listed in a scrollble window. These errors are hot-clickable with auto-jumping to offending code. The inclusion of find and replace text dialogue boxes in both the code and callback editors will also help in editing and debugging your code.

Extended Widget Support

Gauges, Progress Bars and TTY Windows that were introduced into GINOMENU v5.0 are now supported in Studio v3.0. In addition to this Multiple Document Interface (MDI) applications can be created.

The default values for all widgets used on creation can be changed to avoid multiple editing later on in the development cycle.



The forms that you create within Studio can now be saved separately so that they can be imported into other projects.

Help

The manual has been completely rewritten with a tutorial chapter. This is included as on-line help. There is also Context-sensitive F1 Help in all dialogues.

GINOMENU v5.5

Property Lists

In the development of GINOMENU Studio v3.0, a new widget type was required to improve the way the properties of widgets were controlled. This has been called a property list, which is a two column scrollable list allowing the user to see a list of properties and their values.

Cell Type	Value
Width	50
Height	20
Background Colour	0
Text Colour	0
Justification	GLEFT
Border	0
Display Flag	GEDIT
Sign Type	GDEFAULT
Entry Format	GDECIMAL
Field Width	8
Decimal Places	2
Cell Text	0.00

The values can be editable and are by default, text values. In addition to this, combo boxes or numeric entries can be adopted by entries in the list to allow a variety of input methods. If the entry is more complex than this, a callback can be attached to the list in order to allow the application to display a dialogue box to help with the selection. In this case an additional button is displayed when the entry is selected.

The property list automatically recognises HTML hex colour strings in the format "#RRGGBB". The list will display this string in addition to a coloured box reflecting the chosen colour.

To support this new widget, a set of commands have been released to add, remove, enquire and set the property names and values. Features to count and search the property list are also included.

Calendar Display

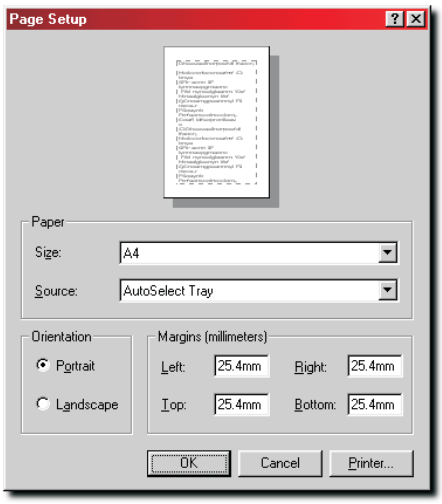
The standard windows calendar is now callable to display a date. This widget is supported by commands to set and enquire the currently selected date. By default one month is displayed. If the widget is enlarged more than one month can be displayed.

Floating Toolbar Support

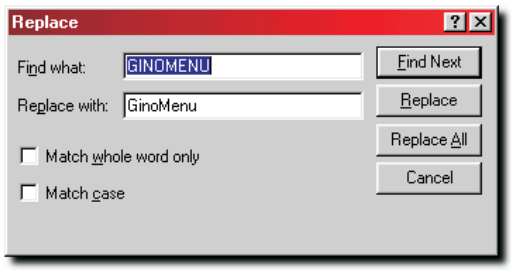
These toolbars were first introduced with v5.0. To support them further is a set of commands to enquire and set their position. They can also be made to be hidden by user or application request. These commands were added to increase the users ability to customise the layout of their application. You should be aware of the users desire to do this and engineer your application to record the position, size and visibility of toolbars, so that you can maintain this setup when the application is restarted at a later session.

Text Printing Facilities

Although the ability to print text has been available for some time in GINOMENU, it has relied on the user setting up the printer via the standard printer set up dialogue box. With v5.5, the application can set up the printer with such properties as paper size and orientation.



The facility for WYSIWYG display of Rich Text has been included with the introduction of page set up commands. This can be done indirectly by dialogue box or by the application. The page size and margins will affect the word wrap allowing you to see what the document will look like when printed. There is now a method of controlling page throws in documents that allows them to be added or removed.



Rich Text Facilities

In addition to the page set up facilities above, find and replace text operations have been added. These can be done from the application or via standard dialogue boxes controlled by the user. As text which has been selected by the user can now be returned as a string, it can be used to pre-fill the find text dialogue box.

Text formatting is extended with the superscript and subscript styles and if you have the latest version of the Windows file "riched20.dll", full justification is possible. In addition to this is the standard font formatting dialogue box.

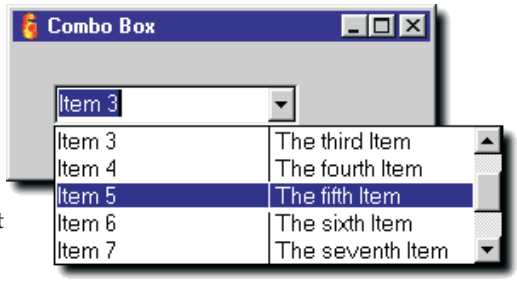
New Callbacks

Two new callbacks have been added. The first is the double click callback which operates within the global system timing. The other is a hotkey select callback which is triggered when a hotkey is pressed but before it is released.

Extensions to the existing callbacks have been included which are documented in a completely re-written callback section in the user manual. This includes more change callback functionality and an implementation of the mouse wheel as a key callback.

Other Features

- Change of widget tabbing and automatic positioning order
- Tracking position enquiry and setting
- Resource file string storage
- Interactive widget position and size editing
- Graphics Frame cursor positioning
- Table box column removal
- Two column combo boxes

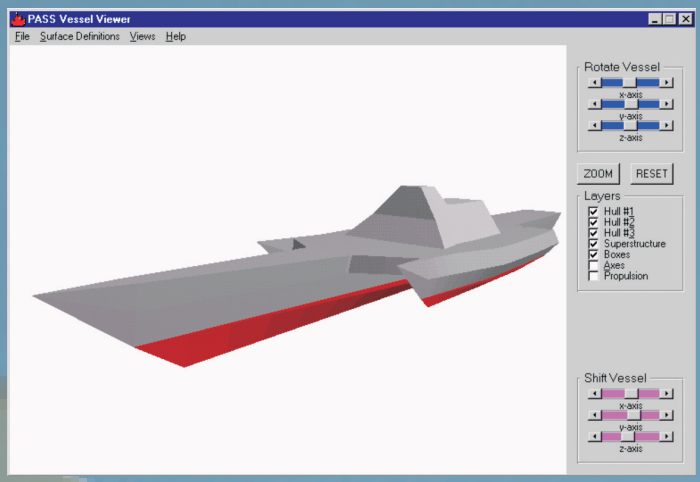


BLA uses GINO for PASS VISUALIZATION

By Jeff Cullina, Ocean Engineer *(jeffrey.cullina@cdicorp.com)*

Band, Lavis & Associates (BLA) is an engineering firm offering a comprehensive range of conventional and advanced marine engineering and naval architectural services. The company is a division of CDI Government Services and provides considerable technical support to the U.S. Navy and Coast Guard providing ship and subsystem design, prototype construction, engineering, analysis and dynamic finite-element structural analysis.

The primary purpose for GINO within BLA is to represent the geometric output provided by PASS (Parametric Analysis of Ship Systems), by way of the Vessel Viewer. PASS is a physics-based computer aided design synthesis model that permits simulation-based design for Displacement Monohulls, Planing Monohulls, Semi-Planing Monohulls, Catamarans, Trimarans, SWATH and Semi-SWATH. For each hullform type, the model permits whole-ship design trade-offs to be examined with respect to acquisition or operating cost and ship performance. The Vessel Viewer originally included an accurate depiction of geometric hull offsets as a wireframe with animation controls. It now generates the hull in wireframe and solid modes and lines drawings (a set of drawings specific to naval architecture), all in a GINO OpenGL environment. Both GINOMENU and the full GINO library are utilized giving GUI capabilities with superb drawing results. With the primary goal complete, the Vessel Viewer is now progressing to modelling subsystem-level components such as decks and bulkheads, major propulsion components (propellers, rudders and major shafting), machinery spaces (engine room), and major structural components (longitudinal and transverse stringers, accurately depicting beam shape, size and quantity).



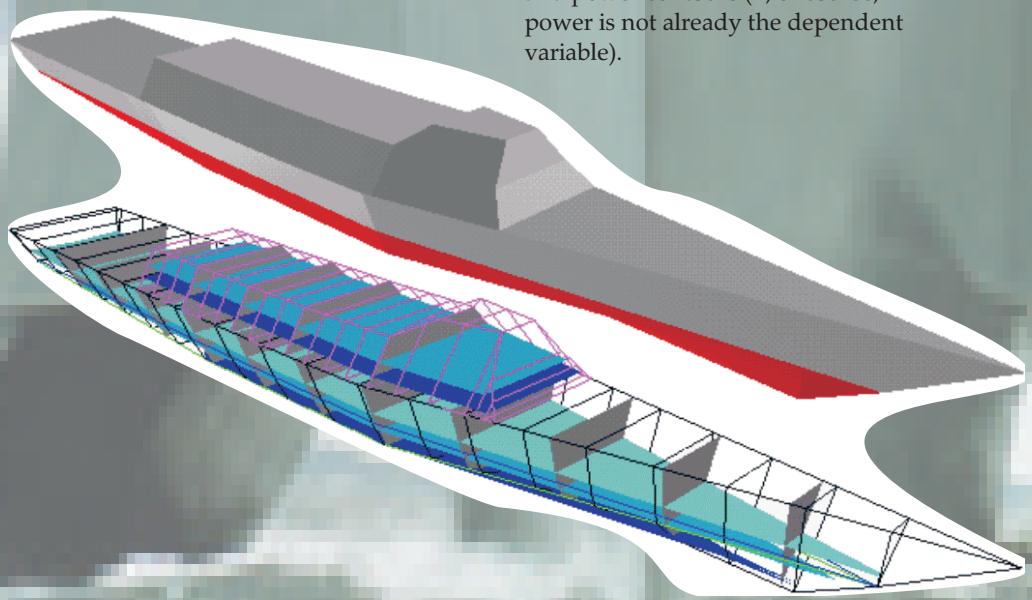
The flexibility of GINO in terms of graphical output has encouraged BLA to pursue the automation of other relatively simple, but redundant tasks. Two-dimension performance plots are now illustrated by a separate executable called within PASS, named PASS Performance, employing GINOGRAF. Users have control over axes and line properties, and may export the plots for reports or any other purpose. The need to open spreadsheets or other mathematical software to generate plots has been eliminated.

GINOGRAF has also been used to illustrate the three-dimensional design space generated by the parametric evaluation performed by PASS. Typically, displacement or power is plotted versus length and beam. Each gridded point represents a design successfully calculated by PASS, which can be further examined at the wish of the PASS user, though typically local minima are of the greatest interest. This executable, Evolution, currently generates the design space (surface plot) accurately, but still awaits controls to overlay significant design information such as a stability limit line and power contours (if, of course, power is not already the dependent variable).

As a natural extension of its GUI capabilities, especially with the availability of the Rich Text Editor, BLA is now in the process of automating the generation of PASS reports through GINOMENU. Outputting data is not an exclusive GINO facility as it can be done with any programming language. It is an outstanding feature, however, that text is initially formatted to the degree of a Rich Text Editor all by automation, and can then be further edited at the will of the user through a GUI before being exported. This eliminates the additional step otherwise required of opening the output in a word processor to perform these actions.

The role GINO plays within BLA has greatly transcended the original intention of BLA, which was simply to model the geometric hullform computed by PASS in wireframe mode. Because of its diversity, BLA has made use of all of the GINO libraries, GINO, GINOMENU, GINOGRAF and GINOSURF in support of PASS, a key software tool frequently used by BLA engineers and naval architects. GINO has served to significantly automate and streamline the design process with PASS while ultimately serving to make PASS wholly independent of additional software as designers progress through the design spiral.

We are excited by the extreme power and versatility of this tool, which is being used every day in our office for not only design, but for assessing the impact of new technologies on naval and commercial ship designs.



GINOMENU Q&A

When I want to save an existing file using gmSaveTextToFile, a message box appears asking if I want to overwrite it. How do I stop this happening?

An optional argument has been added to this routine at v5.5 called gmStatus. Set it to GOVERWRITE to overwrite the file without prompting. If you are using the F77 interface routine TEXSAV from v5.0 and above, set the third parameter to -1 to have the same effect. Remember that this is a return variable.

I would like to use the rich text editor as an internal help file. Does it support hotlinks?

There is a setting that allows a hotlink style to be given to text but this doesn't work very well so it wasn't included in GINOMENU. You can simulate it using existing v5.0 commands.

Firstly, make the hotlink text a particular style, such as blue and underlined. Next set up a change callback on the rich text editor widget. This change callback will report the character position at which the user has pressed. If the text attributes at this position are the same as your hotlink attributes, the user has clicked on a hotlink. It is now up to you to work out the word that was clicked on using enquiry routines.

In v5.5, this operation can be simplified by using the double click callback instead. The action of double clicking a text editor is to highlight the word. Use gmEnqSelectedText to retrieve it if its attributes match those of your hotlinks.

Don't forget to make your editor "read only" if you want to use it as a help file or reference guide.

I've switched to pixel positioning to get more widgets on the screen but the text looks too big. Is there any way around this?

Although you are not using the positioning grid, which is set using gmDefineGuiGrid, it controls the default font used. The default value is 32,24 which represents the number of grid cells within an 800 x 600 pixels area. By changing these values to 36,27 the font will get smaller. When using this remember that the normal font is a bitmap font, only scalable in steps and that it changes the grid cell size which is used internally by GINOMENU. The most noticeable effect is single line text entries changing to multi-line as their height will now be bigger than one grid cell if you've sized them using pixels.

How can I detect that the user has quit a window?

By attaching a general callback to a window, you will be alerted when the user has quit, moved, resized, scrolled, deselected or minimized the window. Using the gmEnqActionState to get the status value, you can find out what has happened to the window. If the status value is 2, the user has quit the window.

What's the best way to change a single property of a widget?

The best way to change a single property of a widget is to use the command gmSetWidgetProp using one of the optional properties listed in Appendix C of the GINOMENU manual. This command will change a single property leaving the others as they were. In many cases it is also necessary to redraw the widget after the change.

How do I create icons for use with GINOMENU?

The icon format is now very complex and is capable of holding different sized icons with different colour depths so it is best to use a 3rd party icon designer. We use Microangelo from Impact Software. It is reasonably priced and it has a fully functional trial version. Details can be found at www.impactsoft.com.

I am creating a child MDI dialogue before calling gmManage and it doesn't display properly. What is going wrong?

MDI is a complex process which is largely shielded from the GINOMENU application writer. The MDI master window needs to be fully initialized before it accept child windows. The initialization is complete after you call gmManage, so add your child windows then. Adding other types of widgets such a toolbars, menubars and status bars can and should be done before calling gmManage.

How do I add an icon into the middle of a toolbar that I've already created?

When you create a new icon for a toolbar it is automatically added as the last icon. This is because the toolbar sorts its widgets into creation order. In v5.5, a new command has been introduced to change the creation order. Using the command gmChangeWidgetOrder, you can set a widget to appear to have been created just before or just after another specified widget. Using this command you can position the last icon added to anywhere within the toolbar. This routine will also change the tabbing order of any widget.

Win 9x/NT/2000/XP Price List

GINO-F Bundle v5.5 (includes GINO-F,GRAF,SURF,MENU)	£1400	$2100	€2200
GINO-F Super Bundle v5.5 (Bundle + Studio + Extended Warranty)	£1995	$2995	€3150
GINO-C Bundle v5.5	£995	$1495	€1575
GINO-C Super Bundle v5.5	£1595	$2395	€2520
GINOMENU Studio v3.0	£300	$450	€475

Win 9x/NT/2000/XP Upgrade Price List

GINO-F Bundle v5.0 - v5.5	£300	$450	€475
GINO-C Bundle v5.0 - v5.5	£300	$450	€475
GINOMENU Studio v2.0 - v3.0	£100	$150	€160

Due to fluctuations in the currency exchange rate, these prices are only a guide. Apply for a quotation to get current prices.

GINO News is the newsletter of Bradly Associates Ltd. While Bradly Associates Ltd. make every effort to ensure that information contained in GINO News is accurate, alterations may occur, particularly in the availability or technical specification of products. For this, GINO News is circulated for information only and does not constitute a specification. Readers are advised to contact Bradly Associates Ltd. for further information on any products. All other trademarks where used are acknowledged. This publication may not be reproduced in whole or in part without the prior written consent of Bradly Associates Ltd.
Published by Bradly Associates Ltd. Manhattan House, 140 High Street, Crowthorne, Berkshire, RG45 7AY UK
Tel: +44(1344) 779381 Fax: +44(1344)773168 E-mail: sales@gino-graphics.com URL: www.gino-graphics.com

©2002 Bradly Associates Ltd. All Rights Reserved